Management API

WAZUG 7

Marcel Meijer - 2012-03-14

Marcel Meijer

Senior Architect

@marcelmeijer

MHMMeijer@vxcompany.com

http://www.marcelmeijer.net
http://blogs.msmvps.com/marcelmeijer

MVP Microsoft Most Valuable Professional

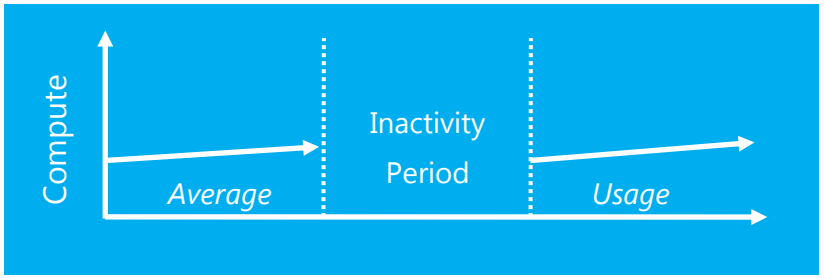
Windows Azure

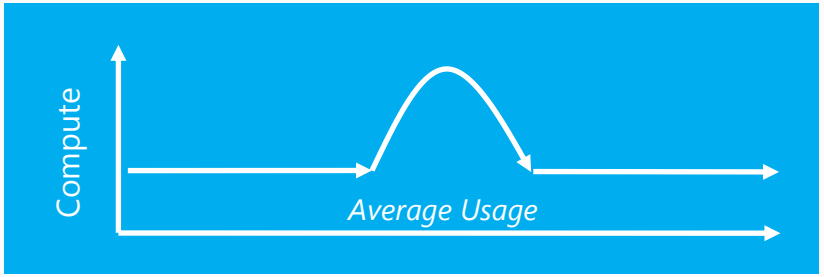
VX Company IT PROFESSIONALS

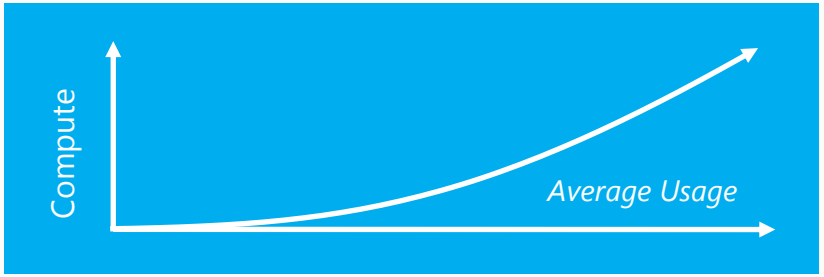
SDN SOFTWARE DEVELOPMENT NETWORK

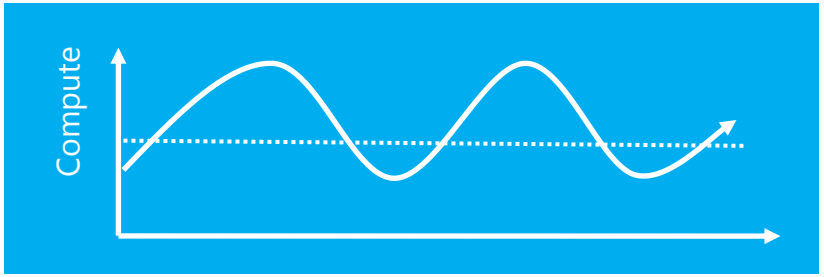

Once upon
a time...

Cloud Scenarios

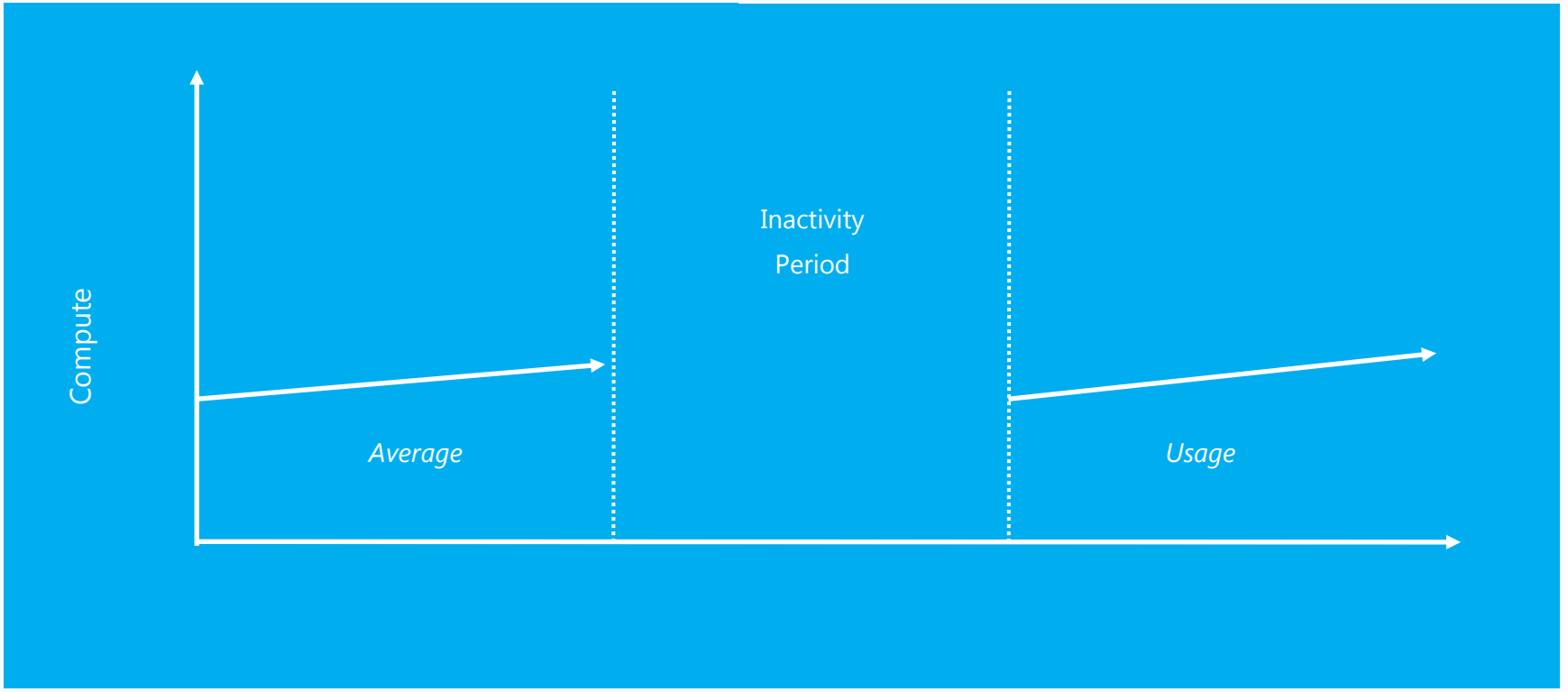






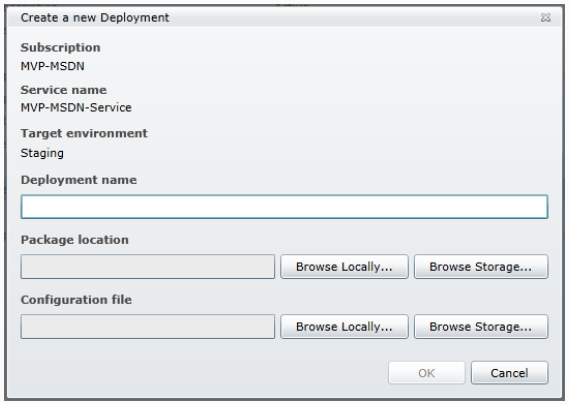


Cloud Scenarios

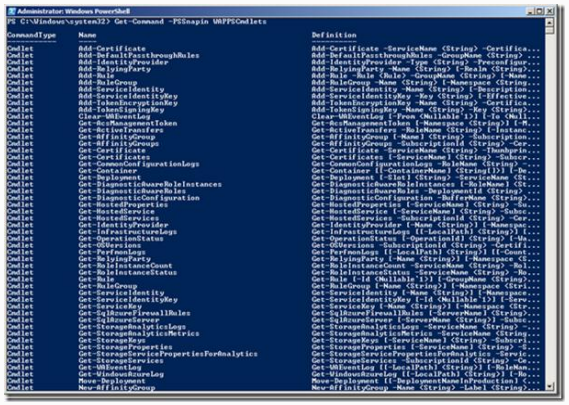


Deployment

Handmatige Deployment via Windows Azure Portal



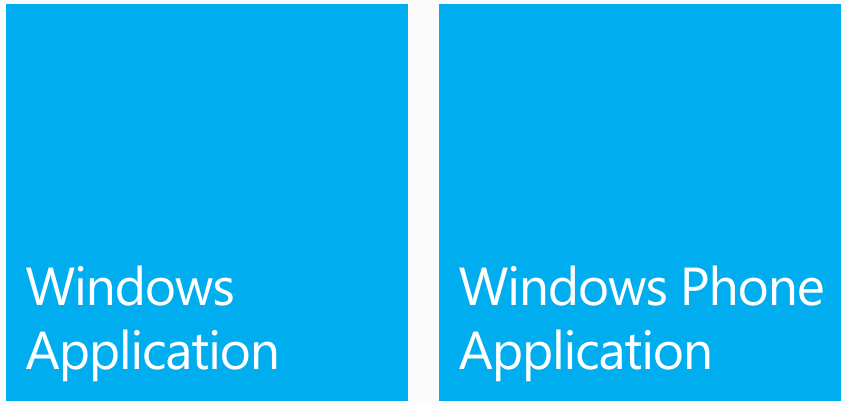
Windows Azure Power Shell Cmdlets



But…

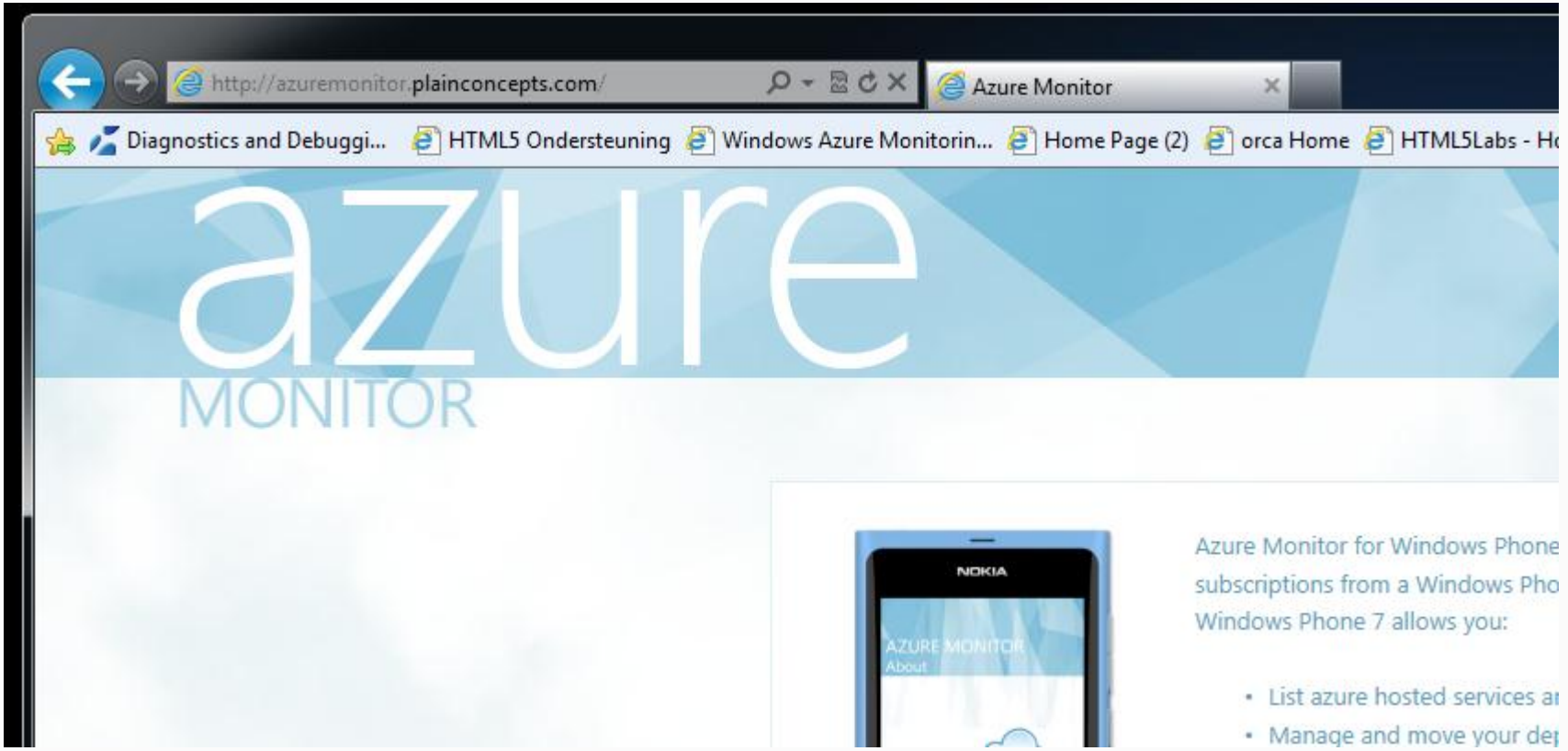
!=

Solution



Windows Azure Managment

Windows Phone



Windows Phone

REST API

Demo

REST API

```
XDocument GetListOfDeployments()
{
    var url = String.Format("https://management.core.windows.net/{0}/services/hostedservices/{1}/deploymentslots/{2}"
        , AzureConst.subscriptionID
        , AzureConst.serviceName
        , AzureConst.stagingOrProduction);

    HttpWebRequest webRequest = (HttpWebRequest)HttpWebRequest.Create(url);
    webRequest.Method = "GET";
    webRequest.ContentType = "application/xml";
    webRequest.Headers.Add("x-ms-version", "2011-10-01");
    X509Certificate2 certificate2 = GetCertifcate();
    if (certificate2 == null)
        return null;

    webRequest.ClientCertificates.Add(certificate2);

    try
    {
        XDocument responsePayload;

        using (HttpWebResponse response = (HttpWebResponse)webRequest.GetResponse())
        {
            Stream responseStream = response.GetResponseStream();
            responsePayload = XDocument.Load(responseStream);
        }

        return responsePayload;
    }
    catch (Exception ex)
    {
        Debug.WriteLine("Error: " + ex.Message);
    }

    return null;
}
```

REST API

```
private async Task GetListOfDeployments()
{
    var url = String.Format("https://management.core.windows.net/{0}/services/hostedservices/{1}/deploymentslots/{2}"
        , AzureConst.subscriptionID
        , AzureConst.serviceName
        , AzureConst.stagingOrProduction
        );

    try
    {
        WebClient wc = new MyWebClient();
        string content = await wc.DownloadStringTaskAsync(url);
        XDocument responsePayload = XDocument.Parse(content);
    }
    catch (Exception ex)
    {
        Debug.WriteLine("Error: " + ex.Message);
    }
}
```

VS11 + C# 5

REST API

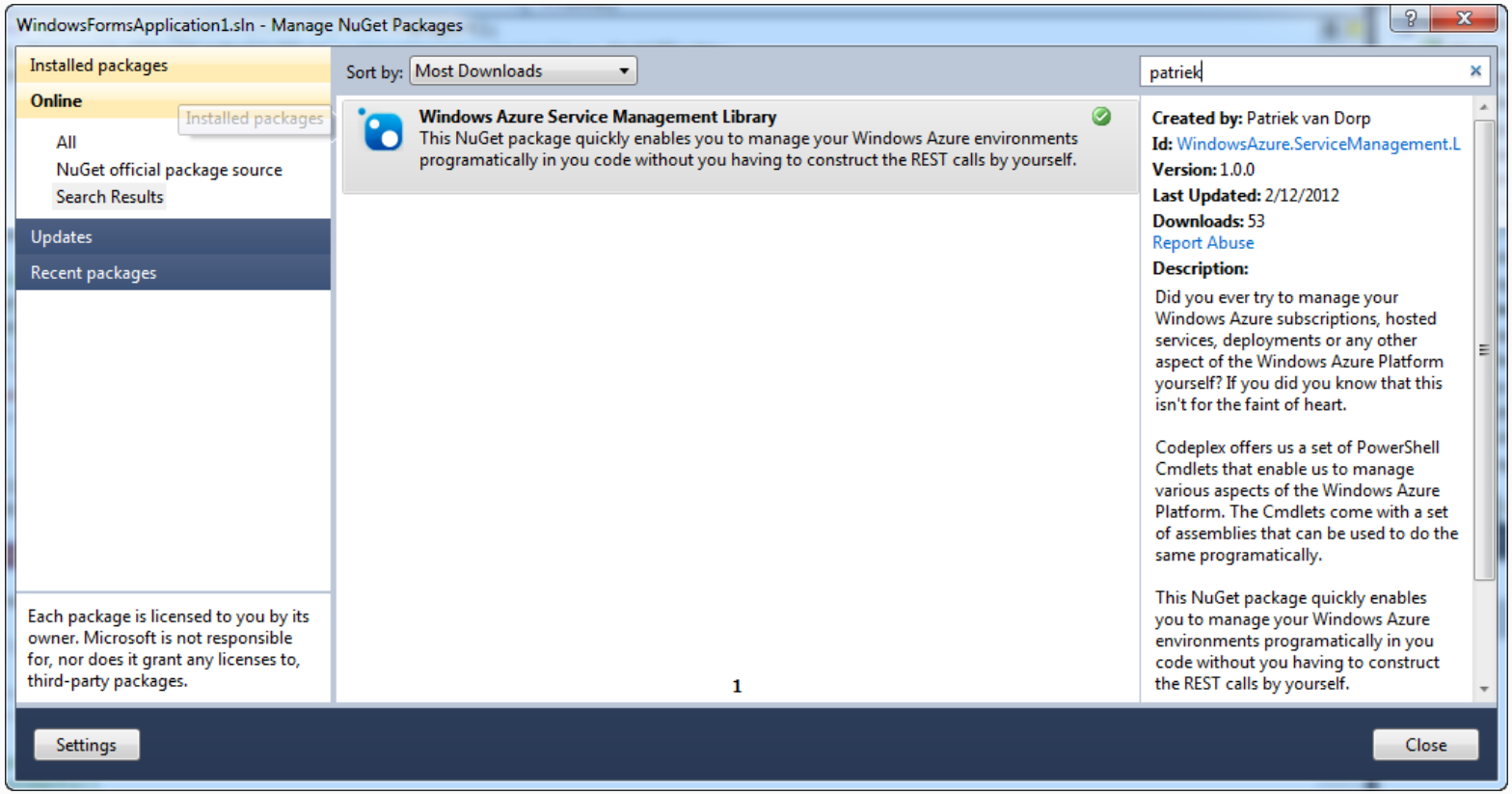
```
public class MyWebClient : WebClient
{
    protected override WebRequest GetWebRequest(Uri address)
    {
        HttpWebRequest request = (HttpWebRequest)base.GetWebRequest(address);
        request.Method = "GET";
        request.ContentType = "application/xml";
        request.Headers.Add("x-ms-version", "2011-10-01");
        request.ClientCertificates.Add(GetCertifcate());
        return request;
    }
    static private X509Certificate2 GetCertifcate()
    {
        var certificateStore = new X509Store(StoreName.My, StoreLocation.CurrentUser);

        certificateStore.Open(OpenFlags.ReadOnly);

        var certs = certificateStore.Certificates.Find(X509FindType.FindByThumbprint, AzureConst.thumbprint, false);
        if (certs.Count == 0)
        {
            Debug.WriteLine("X509 certificate niet gevonden in store.");
            return null;
        }

        return certs[0];
    }
}
```

WA CMDLets DLLs

Demo

WA PowerShell CMDlets

- http://wappowershell.codeplex.com/
- NUGet



DLL's

```
public Form1()
{
    InitializeComponent();

    // Save arguments to variables
    var certificateFile = @"D:\DATA\DOCUMENTS\CERTIFICATEN\MyLaptopCer.cer"; //args[0]
    var subscriptionId = "[redacted]"; //args[1]
    var serviceName = "marcelmeijer"; //args[2]
    var roleName = "MyNewAzureWebRole"; //args[3]
    var slot = "Production"; //args[4]
    var instanceCount = "2"; //args[5]

    // Do the magic
    var managementClient = Microsoft.Samples.WindowsAzure.ServiceManagement.ServiceManagementHelper.CreateServiceManagementChannel(
        WebHttpBinding(), new Uri(ServiceEndpoint), new X509Certificate2(certificateFile));

    var deployment = managementClient.GetDeploymentBySlot(subscriptionId, serviceName, slot);
    //deployment.
    string configurationXml = ServiceManagementHelper.DecodeFromBase64String(deployment.Configuration);

    var serviceConfiguration = XDocument.Parse(configurationXml);

    XDocument xmlDocument = XDocument.Parse(configurationXml);
    textBox1.Text = configurationXml;
```

Wrap up

Wrap up

Certificaat nodig

REST call; zelf parsen van de XML

DLL's, maar niet formeel